

Honeybee Social Foraging for Urban Traffic Control

A Thesis
Presented to
The Academic Faculty

by

Pablo Andrés Ñañez Ojeda

In Partial Fulfillment
of the Requirements for the Degree
Master of Electrical Engineering

Department of Electrical and Electronic Engineering
Universidad de los Andes
January 2009

Honeybee Social Foraging for Urban Traffic Control

Approved by:

Nicanor Quijano, Advisor

Date Approved: _____

*A mi familia, por su amor incondicional. A mi advisor, por su apoyo y orientación, para él
mi profunda admiración y agradecimiento.*

ACKNOWLEDGEMENTS

This work would not have been possible without the support and guidance of my advisor and friend, Nicanor Quijano.

I would like to Andrés Pantoja, Eduardo Ramirez, and Eduardo Mojica, for their help and inputs.

I am grateful to SIT Colombia, which let me use the microsimulator VISSIM.

Also I would like to thank Roselly Pajaro, and Ivan Baquero, (at SIT) for their patience and help in VISSIM.

TABLE OF CONTENTS

DEDICATION	iii
ACKNOWLEDGEMENTS	iv
LIST OF TABLES	vii
LIST OF FIGURES	viii
SUMMARY	x
I INTRODUCTION	1
II TRAFFIC THEORY	5
2.1 Traffic flow models	5
2.2 Road Traffic Light Control Strategies	8
III MODEL AND IMPLEMENTATION OF THE HONEY BEE SOCIAL FORAGING ALGORITHM	14
3.1 Analogies Between the Foraging and the Time Allocation Problem	15
3.2 Algorithm	16
IV SIMULATION RESULTS AND DISCUSSION FOR A SINGLE INTERSECTION PROBLEM	19
4.1 Experiment I	20
4.2 Experiment II	23
4.3 Experiment III	25
4.4 Discussion	27
V STABILITY ANALYSIS	28
5.1 Fluid-Flow model	28
5.2 The Replicator Dynamics approach	31
5.3 System equilibrium points	32
5.4 Phase plane	33
5.5 Discussion	33
VI MULTI-HIVE AND MULTI-SITE PROBLEM	35
6.1 Implementation	37
6.2 Results and discussion	38

VII CONCLUSIONS AND FUTURE WORK 42

LIST OF TABLES

Table 1	Main analogies between the two problems.	17
Table 2	Simulation scenarios.	39
Table 3	Simulation results.	39

LIST OF FIGURES

Figure 2.1	Fundamental diagram. Adapted from [1].	6
Figure 2.2	Speed versus Density. Adapted from [1].	6
Figure 2.3	Phases and allowed movements on a single intersection. The percentage of cycle time where the phase i has right of way (r.o.w) is λ_i with $(\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1)$	9
Figure 2.4	Green waves in MAXBAND. Figure adapted from [2].	10
Figure 2.5	Basic structure for TRANSYT algorithm. Figure adapted from [3].	10
Figure 2.6	Phase controlled with the vehicle interval method, a first approach of MOVA algorithm. Figure adapted from [2].	11
Figure 3.1	Roles, information flow and colony model structure.	16
Figure 4.1	Differences in a stochastic process between a Poisson distribution and a heavy tailed distribution. a) Event distributions over time according to a process that obey to a Poisson distribution, where the event can take place in any moment with equal probability. The horizontal axis represents time, and each vertical line corresponds to an event. Notice that the expected times among events are similar and there are not big waiting times. b) , Expected times for 300 consecutive events. The expected time of each event means one blank space in <i>a</i>). c) , Event distribution over time for a heavy tailed distribution. d) , The expected times for 300 consecutive events in a heavy tailed distribution. Notice the big peaks on the picture. These peaks correspond to big wait times, as shown in <i>c</i>). b) and d) are drawn in different scale because of the difference in the maximum values of the expected time. However, the two distribution have the same expected waiting time $\lambda = 6$ [4], [5].	21
Figure 4.2	Intersection controlled using the <i>Honey Bee Social Foraging</i> algorithm when the vehicle arrivals are constant. Flow below the saturation point. On top the queue levels for a short simulation time is shown. The profitability function and the position for each bee are plotted in the bottom right sub plot. For the profitability function, the x axis is related to the time (from -1 to $+1$, where $+1$ is the full simulation time and -1 is the start simulation time). On the lower left corner, the clock of the phases is shown. In spite of the fact that the intersection has two stages, in the simulation, we have waiting time phases (phase 2 and phase 4), the phases that get the right of way are 1 and 3.	22
Figure 4.3	The x axes correspond to the simulation time (k). In a) , the state of each phase is shown. In b) , the suitability for each site is shown. Notice how the IFD is reached when the suitability functions converges. In c) , the value for the queues are plotted. Here, we can see why the intersection its over a saturation point. The queues are filled more faster than their full capacity can serve (i.e., r. o. w). In d) , The total number of bees at sites are shown. Notice how the IFD emerges when the suitability of sites 1 and 2 converges at a similar value.	24

Figure 4.4	The x axes correspond to the simulation time (k). In any moment of the simulation, the fixed time strategy gives better results than the achieved for the bio-inspired algorithm for the same sampling time.	25
Figure 4.5	Intersection controlled using the <i>Honey Bee Social Foraging</i> algorithm when the vehicle arrivals follows a Pareto distribution (heavy tailed distribution). The time between events τ is exactly the same for the experiment I . The proposed solution presents a high adaptability to the circumstances, and the load among queues is stable.	26
Figure 4.6	Statistics of the queues values with fixed time control and using social foraging. It shows clearly that the traffic responsive technique is superior.	27
Figure 5.1	Vehicular input (q_{in_i}) and output (q_{out_i}) flows, densities (k_i), mean (u_i) and free speed (u_f) for a single intersection model.	29
Figure 5.2	Simplified modeling of link speed u_i . Adapted from [3].	31
Figure 5.3	Trajectories of the system for some initial conditions.	33
Figure 5.4	Simulation for a single intersection controlled using the <i>Honey Bee Social Foraging</i> algorithm. a) The states of each phases. b) The queue state for each approach. c) The bee allocation (resource allocation). d) The time used for each phase per cycle.	34
Figure 6.1	VISSIM - Visual Basic - Matlab bridge.	36
Figure 6.2	Multi-site and multi-hive simulation scheme.	36
Figure 6.3	Green waves that emerges naturally in the evolutionary algorithm. a), b) and c) are simulation results for the <i>(vii)</i> scenario, and d), e) and f) are simulation results for the <i>(iii)</i> scenario. a) and d) shown the green waves in the lower horizontal link (when the 3^{th} phase for the lower right and left controllers are active at the same time). b) and e) shown the green waves in the upper horizontal link (when the 3^{th} phase for the upper right and left controllers are active at the same time). c) and f) shown the instants when the green waves in both directions (right to left and left to right) are active (when the 3^{th} phase for all controllers are active at the same time).	40
Figure 6.4	Cross correlation function for the 3^{th} phase in the upper left and lower right intersection for the simulation results of the <i>(vii)</i> scenario.	41

SUMMARY

We introduce a novel traffic control strategy based on a social foraging algorithm. The dynamic resource allocation in traffic signaling resembles the foraging allocation problem in a honey bee colony. The analogies between two problems are discussed. We propose an application for a simple and multiple intersection using the *Honey Bee Social Foraging* algorithm, and we show how an ideal free distribution (IFD) emerges. The inputs of a single and multiple intersection problem are modeled as a stochastic process with a heavy tailed distribution. The multiple intersection problem is simulated in VISSIM.

CHAPTER I

INTRODUCTION

The search for a more efficient, safe, and less contaminant transport system, for the overload traffic networks in modern cities, leads the research on intelligent transport systems (ITS). The efficiency of the ITS depends in most of the cases on the efficiency and suitability of the control strategy used. For this reason the ITS must exploit developments in areas such as telecommunications, computer science, and process control [3]. Traffic congestion appears when too many vehicles attempt to use a common transportation infrastructure with limited capacity. In the best case, traffic congestion leads to queuing phenomena (and corresponding delays) while the infrastructure capacity is fully utilized. In the worst (and far more typical) case, traffic congestion leads to a degraded use of the available infrastructure (reduced throughput), thus contributing to an accelerated congestion increase, which leads to further infrastructure degradation, and so forth [3]. The traffic control in urban networks is commonly implemented by traffic light signals. Using conventional control methods, developed in the 70's and 80's, the modern metropolitan areas suffer the consequences of low performance of these methods under saturation conditions. Some of these strategies are unable to avoid the degraded use of the road network infrastructure (reduced throughput). Under this scenario, the pollution and latency levels are increased, as well as, the safety on the roads is diminished.

The traffic light signal scheduling is a typical dynamic resource allocation problem. This problem is commonly formulated as a combinatorial optimization problem. The basic problem faced by these formulations is due to the presence of discrete variables that require exponential-complexity algorithms for a global minimization. In fact, optimization methods based on models such as OPAC [6] employs complete enumeration (assuming integer switching times) while PRODYN [7] and RHODES [8] employ dynamic programming. Due to the exponential complexity of these solution algorithms, the control strategies (though conceptually applicable to a whole network) are not real-time feasible for more than one intersection.

The dynamic resource allocation problem is solved using a bioinspired technique that is very suitable in this kind of problems. The traffic light signal scheduling is formulated as a dynamic resource allocation problem. Then, a bioinspired algorithm (based on how the animals distribute themselves in some environment in order to optimize their nutrient consumption rate) is adapted to solve the green time allocation problem without the necessity of traffic models. This property makes the solution feasible for large scale networks in a real-time and distributed control operation. The green time allocation in semaphored intersection resembles the foraging problem. The main resemblance between these two problems is the resource allocation aspect.

Bioinspired techniques have been implemented in a variety of engineering problems similar to urban traffic control problems. In general, biomimicry is more apt to succeed if there is a close homology between the biology and the application, where the main aspects of the problems make a good match between them [9]. For example, Dorigo *et al* [10] mimic ant foraging behavior and have used their algorithm in the solution of classical optimization problems (e.g., discrete combinatorial optimization problems) and, in engineering applications (e.g., [11], [12]). In the other hand, some honey bee social foraging algorithms have been successfully implemented in engineering static resource allocation problems for a traffic control problem [13]. From a dynamic resource allocation point of view, in [14], and [15] a novel control strategy is used for a multizone temperature control problem, based on a *Honey Bee Social Foraging* algorithm. These works, highlight the relationship of one type of swarm intelligence algorithms [16] with an evolutionary game theoretic perspective, rather than optimization as it is found on most of these type of bioinspired applications (e.g., [17], [18]). Also, the authors have found some relationships with a very useful concept on behavioral ecology (i.e., the ideal free distribution IFD [19]). The IFD concept has been used to analyze how the animals distribute themselves across different habitats or food patches. These habitats have different characteristics (e.g., one habitat can have nutrient input rate bigger than the other). Nonetheless, the animals tend to reach an equilibrium point when each animal has the same level of suitability in the environment (e.g., nutrient consumer rate). The ideal term means that the animals can sense perfectly the quality of the different habitats and so they search the habitat more

suitable, and the “free” term means that the animals can go to any habitat. The IFD emerges for a single beehive when the foragers are assigned to the nectar patches (e.g., see [20], [21], [22]). Even more, if n beehives compete in the same environment for a limited amount of resources the IFD emerges. In [14] the authors show that the IFD is a strict Nash equilibrium in terms of the payoffs to each hive, and a special type of evolutionarily stable strategy. For the n beehives case, the fact that the IFD is an optimum global point means that if we allocate all but $(n - 1)$ individuals at the IFD, then the beehive that is not using the strategy must distribute its workers according to the IFD, in order to maximize its profitability.

Other applications on honey bee social foraging can be found in [23], where the authors analyze the analogies between the resource allocation in a honey bee colony and the server allocation in a center host services with a server ensemble. The center must allocate their servers dynamically amongst services to maximize revenue earned from hosting fees. Also, in [24] the author emulates the most important aspects of social bee foraging [25] for an Internet search engine (“*Tocorime Apicu* search engine”). This search engine seeks and maps out Internet contents from Internet services providers (which host web services). This method establishes routes avoiding congested areas in the Internet.

In this document, a bioinspired algorithm for green time allocation problem, in a single and multiple intersections scheme, is presented. The remainder of this document is organized as follows. First, in Chapter 2 an introduction on traffic theory and control strategies is presented. A particular evolutionary algorithm (the *Honey Bee Social Foraging* algorithm [14]) and the resemblances between the green time allocation problem in urban traffic control and the resource allocation problem in social foraging theory are presented in Chapter 3. In Chapter 4, the implementation and results of the *Honey Bee Social Foraging* algorithm for the dynamic time allocation in a single intersection is presented; then, we model the arrivals to a single intersection as a stochastic process (that follows a Heavy tailed distribution [4]). Also, it is shown how the IFD emerges under a variety of conditions. In Chapter 5, The *Honey Bee Social Foraging* algorithm proposed, and the traffic flow are modeled using the replicator dynamics approach, and a fluid flow model based; later, a comparison between

the simulations is presented. The multi-hive and multi-site problem for a small traffic network is presented in Chapter 6. Here, the simulation of a small network using a microscopic traffic simulator VISSIM controlled by the algorithm proposed is shown. We finally conclude and present some future directions for our work in Chapter 7.

CHAPTER II

TRAFFIC THEORY

Traffic theory studies the road traffic dynamics. These dynamics are related with the interactions between the vehicles in the network. The main characteristics of the traffic networks, such as capacity and availability are function of this interactions.

Nowadays, the traffic control systems implemented can be classified in two classes: fixed time strategies, and traffic sensitive strategies. These strategies can be also classified in isolated or coordinated [3]. Next, the main strategies in these classes are presented.

2.1 Traffic flow models

The main characteristics for the traffic networks (e.g., their capacity and availability) are consequence of the interactions between vehicles. These interactions are stated in various types of traffic theory. These theories are commonly classified in three main groups: *macroscopic*, *microscopic*, and *mesoscopic* models. In the macroscopic approach, the traffic flow could be modeled as a fluid across a long pipe (the highway). In the microscopic models, the single movements of each vehicle are taken into account. Mesoscopic models describe the average speed in a similar way to macroscopic models, but are also considered individual vehicles packed into small groups moving accordingly with the average speed across the network [1].

2.1.0.1 Macroscopic models

In 1955, Lighthill and Whitham [1] described traffic flow as a fluid through a pipe. They study a reasonable crowded, flat, and straight highway where the gaps between vehicles are not very large. In this case, the traffic flow can be considered as continuum, and can be modeled as an imaginary fluid. Some special considerations of geometry of the roads or different driver behavior can be considered. For example, Newell (1965) was the first one to take into account the effect of different

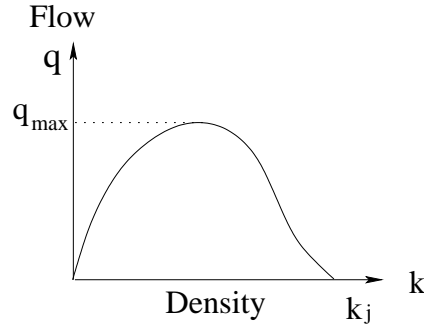


Figure 2.1: Fundamental diagram. Adapted from [1].

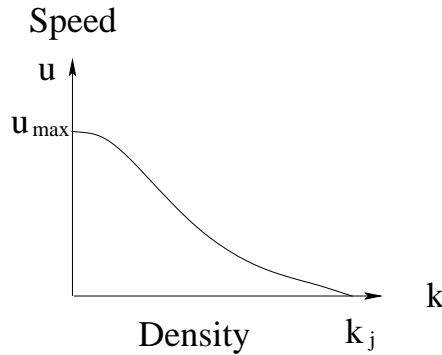


Figure 2.2: Speed versus Density. Adapted from [1].

driver behaviors during acceleration and deceleration [26].

In their dissertation, Lighthill and Whitham study the transient for the flow and vehicular density when the system stable state changes. They assume that vehicular flow q and vehicular density k are related in the *fundamental diagram*, as the one shown in the Figure 2.1. The fundamental diagram can be explained as follows. For vehicular density zero ($k = 0$) there is not traffic flow ($q = 0$). The traffic flow is also zero for some jam density k_j , when the gap between the vehicles is too small and the vehicles stops. These assumptions explains the fundamental diagram shown in Figure 2.1[1]. In this diagram, the relationship between density and flow is shown. When the density increases from zero, the flow increases according to the density. When the density approaches the mean traffic speed (shown in Figure 2.2[1]), the flow reaches its maximum value. When the density surpasses this density, the vehicular flow decreases when the density increases. Then, the jam density k_j is reached when the flow and the speed is zero. In [27] and [28] the vehicle conservation

was proposed. This principle is expressed by the relation $\frac{\partial k}{\partial t} + \frac{\partial q}{\partial t} = 0$, and this is used in Section 5 for a single intersection model. The main restriction for this relation is that Lighthill assumes an infinitely long way, flat and straight. The variations in the geometry of the roads has an impact in the behavior of the traffic flow. The existence of hills and curves introduce variations in the flow and local density, but this variations can be expressed as multiples road sections with multiples flow and density relations [29].

In addition to the geometry another important consideration can be taken into account, i.e., the driver behavior can affect the flow-density relation. Newell (1965) [1] considers the effect of the driver behavior during the acceleration and deceleration. Newell realizes that in general, the drivers allow bigger gaps between vehicles during the acceleration than in the deceleration process. This driver behavior generates low flow values during the acceleration for the same density value. Newell assumes that the properties for the traffic during the acceleration and deceleration are different, and defines this effect as the acceleration bottleneck. This asymmetry affects the velocity of the vehicles and in consequence affects the flow-density relation. This acceleration-deceleration asymmetry can be observed when a vehicle group pass trough a curve. The vehicles diminish its velocity in when they are approaching to the curve, later, when they pass the curve accelerate to its previous velocity. This shows how the road geometry affects the driver behavior and the flow-density relation [26].

2.1.0.2 *Microscopic models*

Wiedemann's model (1974) [30] is the most classical example in these driver behavior models. This model assumes that a driver can be in one of four driving sub-models: (i) *free driving*, where there is no influence of others vehicles observable; (ii) *approaching*, where the diver adapts his velocity to the lower speed of a preceding vehicle; (iii) *following*, where the diver follows a preceding vehicle without any conscious acceleration or deceleration; and (iv) *braking*, where the driver applies medium to high deceleration accordingly with their surroundings.

2.2 Road Traffic Light Control Strategies

The congestions in urban road networks extended in both space and time are caused by the fact that the demand may temporarily exceed the capacity of some specific links. This limited and temporary congestion leads to an infrastructure degradation when no suitable control strategies are employed to avoid this devastating evolution [3].

Control strategies may be classified according to the control scheme implemented in the traffic signals. When the control law is independent of the variables of the network (on real time), the strategy is called *fixed-time* (open loop control). On the other hand, when the control policy uses real time information (feedback), the strategy is called *traffic-responsive* (closed loop control). Also, these control strategies can be classified according to their relationships between near nodes on the network. Thus, the strategy can be classified as *isolated* or *coordinated*.

2.2.0.3 Fixed-time strategies

Using historical data of flow and traffic density, the fixed time strategies are set out for long periods of time (e.g., rush hour), where the variables are assumed to be constants. Isolated fixed-time strategies are only applicable to undersaturated traffic conditions. The optimization problem is solved for these algorithms setting the optimal parameters (splits, cycle time, and staging). The most important algorithms are described below.

- **Isolated strategies.** Nowadays, the majority of the control traffic systems operate under pre-specified fixed-time schedules. The main examples under this class are SIGSET and SIGCAP [31], [32]. Let m be the number of pre-specified phases. SIGSET sets out each phase duration according to $\lambda_0 + \lambda_1 + \dots + \lambda_m = 1$, where λ_i is the percentage of cycle time of the i^{th} phase, and $\lambda_0 = L/c$ is the total lost time/cycle time, where c is the cycle time and L is the total lost time in a cycle. In Figure 2.3 is shown a simple intersection, and associates it with the sequence for the phases. In order to avoid the formation of a queue, a constraint must be satisfied: $s_j \sum_{i=1}^m \alpha_{ij} \lambda_i \geq d_j \forall j$. Here, s_j and d_j are the saturation flow, and the demand for the stream j , respectively. In addition, α_{ij} is equal to one if the stream j has the right of way (r.o.w.) at stage i , and zero elsewhere. The constraint requires that the demands of each

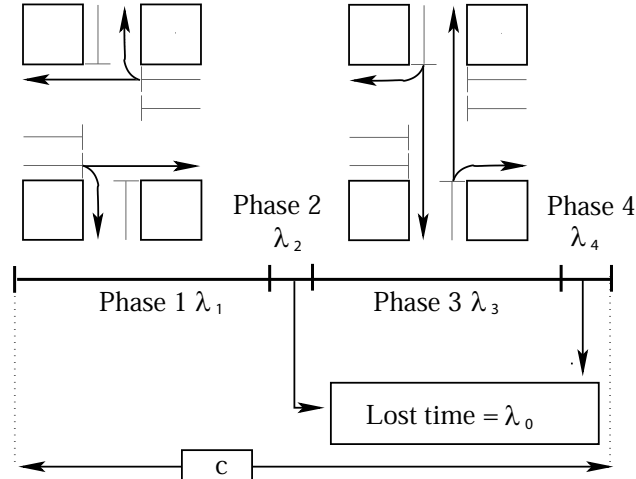


Figure 2.3: Phases and allowed movements on a single intersection. The percentage of cycle time where the phase i has right of way (r.o.w) is λ_i with $(\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1)$.

stream should not be higher than their maximal capacity (saturation flow). In [33], Webster proposes a non linear function for the total delay spent. This function is used in SIGSET as an objective function. SIGCAP is similar to SIGSET, but instead of minimizing the queue size, it maximizes the total flow on the intersection.

- **Coordinated strategies.** In this class of strategies, the most representative strategies are: MAXBAND (*Maximal Bandwidth*), where the green waves are introduced [34], and TRANSYT (*Traffic Network Study Tool*) [35]. Because of its performance under no saturation conditions and that they do not require any kind of instrumentation for real time measurements, these are the most typical strategies used around the world. MAXBAND maximizes the total number of vehicles that can travel within a given speed range without stopping at any signal (green wave), as it is shown in Figure 2.4. The problem faced by MAXBAND consists in one principal road with several intersections with n semaphored signals ($Signal_1, \dots, Signal_n$). The cycle and the green allocation times are calculated according to the demands for each lateral street. The problem consists in placing the offset in the green times for the different intersections across the principal road in order to maximize b (see Figure 2.4). For an appropriate problem formulation, it is necessary to introduce some binary decision variables, which leads to a binary-mixed-interger-programming problem [3], [36].

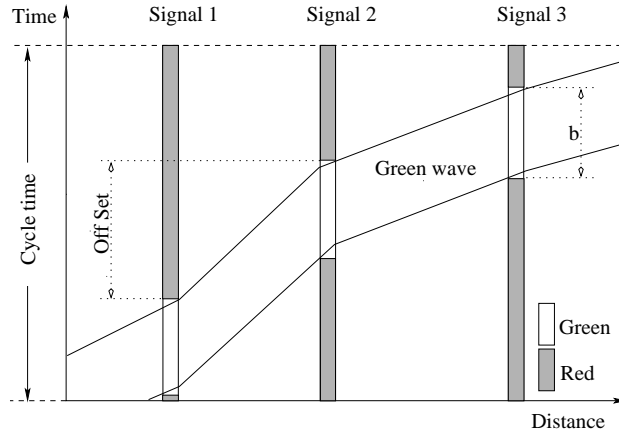


Figure 2.4: Green waves in MAXBAND. Figure adapted from [2].

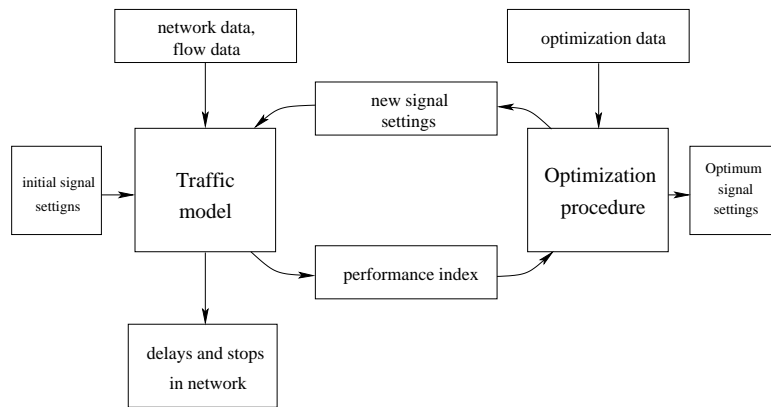


Figure 2.5: Basic structure for TRANSYT algorithm. Figure adapted from [3].

In 1969, Robertson [35] develop TRANSYT (*Traffic Network Study Tool*). In the Figure 2.5, the basic structure for this algorithm is shown. The algorithm is fed with an initial settings of green time and phases for each intersection, the minimum green time for each cross road, offset, and time cycle. The network and flow data are related with the network geometry, the saturation flows, the link travel times, and the constant and known demands. The method proceeds in an iterative way, i.e., for given values of decision variables (control inputs), the model calculates its performance index (e.g., total number of stops, total time spent, service level). The optimization procedure (gradient) introduces small changes in the decision variables and the performance index is re-calculated, and so forth, until a (local) minimum is found.

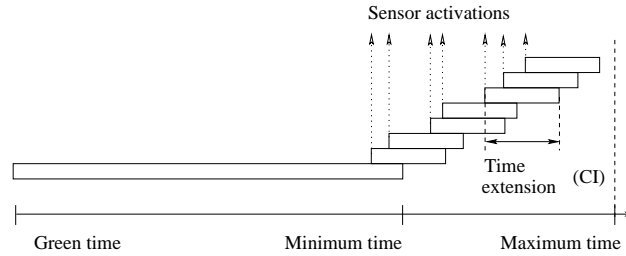


Figure 2.6: Phase controlled with the vehicle interval method, a first approach of MOVA algorithm. Figure adapted from [2].

The main lack of fixed time strategies is that they are based on historical data. This approximation can not be accurate for the highly changing demand dynamics (e.g., during a special event, car incidents can affect the traffic in a unpredictable way). For this reason, the traffic-responsive control strategies are potentially more efficient, but also more costly, as they require the installation, operation, and maintenance of a real-time control system (sensors, communications, central control room, local controllers) [3].

2.2.0.4 Traffic-responsive strategies

- Isolated strategies.** These strategies use real time traffic and density measurements. This information is used to adjust the green allocation setting. Each intersection is adjusted in a separately manner. One of the most used strategies in this classification is the vehicle interval method. In this strategy, for each phase, a minimum green time is assigned. If there is no vehicle passing over the sensors, automatically passes to the next phase. If a vehicle is detected, a critical interval (CI) is increased in the active phase. This happens until the phase reaches a maximum green time. If no vehicle is detected during the CI, the algorithm proceeds to the next phase, as it is shown in Figure 2.6. A more elaborated version of this kind of strategies is proposed by Miller [37] in the MOVA strategy. There it is evaluated every T seconds the time gains and losses in all approaches if the next phase switch is postponed kT seconds.
- Coordinated strategies.** These strategies show a good performance in no saturation conditions, but they require the instrumentation to measure network variables such as flow and

vehicular density in real time [38]. These techniques calculate the control parameters according to the current traffic conditions. In this classification the most important algorithms are:

– SCOOT: (Split, Cycle, and Offset Optimization Technique) [39]. It is considered as a traffic sensitive version as TRANSYT. The algorithm uses a similar philosophy of TRANSYT, but instead of solving the optimization problem based on historical data, SCOOT uses real time information in order to solve the optimization problem in real time. SCOOT has three key principles [40]: (i) *cycle flow profile (CFP)*, this aspect is similar to the one used in TRANSYT, but in this case they are updated every few seconds. In the case of TRANSYT, the accuracy depends on the data about the average flows, saturation flows, cruise time and some other variables. In the case of SCOOT this process is automatic and carried out in real-time; (ii) *update of online model and queue estimation*, the queue estimation is carried out in the same way as in TRANSYT. The calculation of the queue is carried out every few seconds while the data collected from the sensors is collected every second. The detectors are placed upstream and downstream of the intersection; (iii) *incremental optimization*, SCOOT uses an elastic coordination that stretches or shrinks the coordination plans to match them to the detected CFP. This process is carried out to evaluate the effects of modifying up or down the length of the cycle in a series of frequent but small increments. For some intersections the decisions are not suitable but that is compensated with the large majority.

- Optimization methods based on models: In this category we find OPAC [6], PROLYN [7], CRONOS [41], RHODES [8]. These algorithms are known as RT-TRACS (*Real Time Traffic Adaptive Control System*). Using a network model and based in the current control network configuration, the algorithm calculates the optimum configuration for a time horizon.

Nowadays, the control strategies commonly used (e.g., optimization methods based on models) face the problem that require exponential-complexity algorithms for a global minimization because to the presence of discrete variables. This makes the implementation not feasible in large scale networks. For this reason, a novel approach is presented. Here the traffic problem is formulated as

a resource allocation problem, and the strategy proposed here, does not require a traffic model such as some strategies described in this section.

CHAPTER III

MODEL AND IMPLEMENTATION OF THE HONEY BEE SOCIAL FORAGING ALGORITHM

Evolution has adjusted the behavior of the bee colonies to solve optimization problems, where the complexity of them overcomes the capacity of a single individual. Nevertheless, the social behavior accomplish the task of the problem. This is the strength of the mass intelligence [25]. In this case, the problem that arises is that a bee colony must organize its workers into different nectar sources around its habitat, to supply its necessities.

In [14] and [15] the authors describe the bee allocation as follows. Let B be the total number of bees in the hive, which are allocated at N different sites. The constraint $\sum_{i=1}^N x_i = B$ must be satisfied. Let $s_j = \frac{q_j}{x_j}$ be the suitability of foraging at site j , which is a measure of the colony return for harvest q_j (nutrients) by x_j bees. With this representation, the sites represent a choice for the hive, with the site degrading in suitability via the visit of each additional bee. In other words, s_j represents the return from site j per the amount of bees allocated to site j . According to [14], we take a perspective that a single hive wants to allocate its foragers in order to optimize its payoff (which is a correlate of the suitability) by minimizing their investment. The proportion of the total number of foragers at site j is $p_j = \frac{x_j}{B}$ for all j . Then, it will maximize its return of investment by minimizing its losses.

If q_j is in units of nutrients per second, we can make an analogy between nutrients and vehicles per second (traffic flow). In the same perspective, a hive allocates x_j foragers at some site j (in units of “bees”), then, we can make an analogy between bees and time allocated at some phase j (in “time units”). Summarizing, the *Honey Bee Social Foraging* tries to allocate its foragers in order to optimize its payoff by minimizing their investment. In the same way, the green time allocation problem on a single intersection tries to set the total green time in order to optimize its performance

by maximizing the flow, i.e., minimizing the time used for vehicles through the intersection.

To model the process of social foraging it is necessary to model the environment and its habitats. Also, we need to model the activities of the bees, i.e., expeditions, nectar discharge, decision of the strength dance, assignation of explorers, recruitment with the explorer/worker dancers, and the interaction with other beehives [15].

The bee colony model gets and processes the information as shown in Figure 3.1. Each one of the bees can take different roles, and thus uses and transmits the information for different purposes¹. In short, the workers and explorers visit the nectar supplies and returns with the nectar and an estimate of their energetic profitability. When the nectar is downloaded at the beehive, the workers interact with other bees (observers and unemployed). This interaction provides a feedback of the total amount of nectar at a given site harvest for the total number of workers. The estimate of the profitability by each worker of the beehive depends on the time that this experiment takes (e.g., if the time is longer, longer will be the estimated profitability). The workers then decide whether to stop harvesting in that patch (and become unemployed), or if they recruit more workers for the site. Observer bees are recruited in different patches according to the dance strength of the dancers. Dancing decision depends on the perceived profitability and a threshold with the rest of the hive. If the reader is interested in all the details of this model we suggest that he/she reads [25] for the best biological approach, or [14]-[15], which has an engineering perspective.

3.1 Analogies Between the Foraging and the Time Allocation Problem

The main analogies between a single intersection problem, and the honey bee social foraging problem are summarized in Table 1. One of the most important shared characteristics is the relation between the quantity of resources assigned to one place and the obtained suitability. If the amount of bees assigned to some site exceeds the number required, overall payoff is degraded, the same occurs with the green time allocation. If the time assigned to some phase exceeds the time that is required, overall performance is degraded. Network saturation is generally caused by traffic flow

¹The model takes into account four possible roles: explorer, forager, observer, and unemployed.

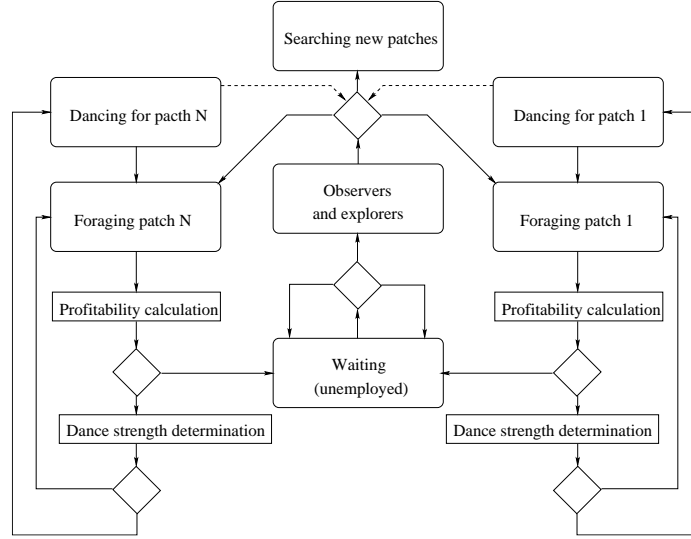


Figure 3.1: Roles, information flow and colony model structure.

fluctuations and excessive demands (phenomena which are usually temporary and local). It is highly desirable to prioritize those intersections that are close or have surpassed the saturation point. In the nearby intersections, where levels are lower, the load must be distributed. Furthermore, the efficiency on the network became balanced.

On a single intersection, as the one showed in Figure 2.3, the IFD seems promising in the search of the optimum time configuration. It is important to recall that the time adjustment can be done dynamically (traffic responsive). Here, the capacity of the honey bees for solving optimization problems in highly changing environments (similar to the experienced in urban traffic networks) is very useful.

3.2 Algorithm

The social foraging algorithm, which is adapted from [14] and [15], is used to assign green time in a simple intersection (simulated in Matlab), as shown in Figure 2.3 and in a multiple intersection (simulated in VISSIM) as is shown in Figure 6.2. In Figure 2.3, the simulated intersection have two stages (phase 1 and phase 3) and its respective lost time (λ_0 , which corresponds to transition phases 2 and 4). The semaphore has a minimum time for each stage (restriction of the problem).

Table 1: Main analogies between the two problems.

Social bee foraging	Traffic control
An ensemble of N nectar foragers allocated in M nectar patches	An ensemble of N time units allocated in M semaphore phases
Blooming / dearth patches spatially distributed	Changing traffic demands on spatially distributed intersections
A group of foragers harvest nectar at some specific patch	A group of time units is assigned at some specific semaphore phase
Each patch is associated with variable rates of quality (sugar concentration & profitability)	Each phase is associated with vehicles in a queue, and the total wait time
The total amount of harvested nectar by a single bee depends on the number of bees at the same patch and on the quality of the patch	The traffic flow during some specific phase per time unit depends on the length of the queue and on the total time allocated in the phase
Variable rates of blooming and nectar quality	Variable demands on intersections

For simplicity, each bee represents a amount of time equal to one simulation instant, which is what it takes a bee expedition. The events in the simulation are measured with respect to this quantity, e.g., a vehicle might waste five instants (bees) to evacuate a row when the semaphore is green. The main objective of the control algorithm is to decrease the queues for each intersection, $Q_1(k)$ and $Q_2(k)$, $k = 1, 2, \dots, N_e$, where N_e is the total number of bee expeditions (N_e simulation instants), and the algorithm has full knowledge of the traffic variables (queue size Q_j , flow q_j and traffic density k_j). Each phase is associated with an habitat i in a simulated environment. The number of bees in the habitat i at expedition k (simulation time k) is $x_i(k)$, so $\sum_{i=1}^N x_i(k) = B$ for all k , which corresponds to the total available time to assign at all phases at simulation time k .

The foragers are assigned to a forage site, and each site is related to the phases, e.g., the site i represents the phase i . The total number of bees at site i represents the total green time at phase i , $t_{phase_i}(k)$, i.e., $t_{phase_i}(k) = t_{min_i} + x_i(k)$, where t_{min_i} is a minimum time required. Hence, the total green time that phase i at simulation time k can experiment is bounded by $t_{min_i} \leq t_{phase_i}(k) \leq t_{min_i} + B$.

The suitability can be a queue, flow or vehicular density function, i.e., the suitability as a function of the vehicular flow q_{in_i} and the saturation flow S_i ,

$$s_i(k) = \min \left(1, \frac{q_{in_i}}{S_i x_i} \right)$$

where the \min function depicts the saturation of the i^{th} queue, flow or density sensor.

The algorithm used in the implementation is adapted from [15]. The core of the algorithm is used to control a single intersection. Next, in Algorithm 1, a pseudo code is shown.

Algorithm 1 *Honey Bee Social Foraging Algorithm for Urban Traffic Control*

Require: $\Theta_i(x, y) = (0, 0), i = 1, 2 \dots B$ //All bees starts at beehive, that is located on $(0, 0)$

Ensure: $\mathcal{E}, \mathcal{F}, \mathcal{U} \in \emptyset \wedge \mathcal{O} = [1, 2, \dots B]$ //The sets of explorers \mathcal{E} , foragers \mathcal{F} , and unemployed bees \mathcal{U} are empty. All bees are initially active observers

$t_R = 0, k = 0$ //The semaphore clock and simulation time are initiated

for $k = 0$ to Ne {Let Ne the total expeditions} **do**

for $i = 1$ to B **do**

for $j = 1$ to N , {were N is the number of sites} **do**

 Bee location and counting up F_{ind} for each zone j

end for

end for

 Phase time updating, and determination of t_R

 Queues updating (microscopic simulation of the queues for the simulation time k , using previous states and t_R)

for $j = 1$ to N **do**

S_j Determination of the suitability for each site according to the queues and the number of bees at j

end for

for $Bee_i \in \mathcal{F} \cup \mathcal{E}$ **do**

$J_{f_i} = J_f(\Theta_i, F_{ind}, S_j \forall j)$ //Explorers and foragers sample the profitability $J_f(\cdot)$ of their previous locations, the number of foragers per site, and the suitability for each site

if $J_{f_i} > \epsilon$ **then**

Bee_i is removed of $[\mathcal{F} \cup \mathcal{E}]$

$Bee_i \in \mathcal{O}$

end if

L_i^j Dance strength determination. //Employed foragers and successful explorers may dance for their sites

end for

 Given a small probability, some bees at \mathcal{O} become explorers

for $Bee_i \in \mathcal{O}$ **do**

 Recruit the unemployed bee by some employed one. According with the dance strength L_i^j , and using the roulette well analogy, the Bee_i is assigned to some forage site j

end for

end for

CHAPTER IV

SIMULATION RESULTS AND DISCUSSION FOR A SINGLE INTERSECTION PROBLEM

Three experiments with two fundamental purposes are proposed. The first one studies the performance of the algorithm in an environment where the habitats q_i are constants (i.e., the arrival to the q_1 and q_2 queues are constants). The second and third experiments focus on the comparison between the performance of the algorithm proposed with the performance of a fixed time strategy. The performance of both algorithms are evaluated as a function to the average of the queues using Monte Carlo simulation. Because the problem in these experiments studies a single intersection, the arrival of vehicles does not depend on the state of the control phases (besides, there are no more intersections which affect these variables). That is why in the experiments *II* and *III* (sections 4.3 and 4.2), they are modeled like a stochastic process.

In the first experiment, two possible scenarios are simulated (section 4.1). First, the system is evaluated when the flow at each approach is lower than the saturation flow. Later in the same experiment, the system is evaluated when the flows are bigger than the saturation flow. This experiment is presented in order to illustrate the behavior of the dynamic cooperative foraging process in two different scenarios. The second experiment (section 4.2) studies the behavior of the intersection when it is controlled with a fixed time strategy. Here, the vehicle arrival at each queue is a stochastic process that follows a Pareto distribution (heavy tailed distribution [4]). In this experiment, three hundred Monte Carlo runs are simulated, and on each one of these, one thousand instants are simulated (simulation time). Finally, in the third experiment (section 4.3), the intersection is controlled by the *Honey Bee Social Foraging* algorithm with two hundred bees at one single hive. Here, we use exactly the same random arrivals, number of instants simulated (expeditions), and the same number of Monte Carlo simulations, than in the second experiment.

The dynamics of many social, economics, and technological phenomena associated with the human decisions is assumed to be randomly distributed in time and similar to a Poisson distribution. However, there is evidence that many of these phenomena obey to a non Poisson process, characterized for many events in a short period of time, straight after long periods of wait. The nature of the stochastic processes where the human decision is involved is similar to a “heavy tailed distribution”[4]. This distribution is a consequence of the task selection process based in priorities, where all events or tasks are not treated as equals. In this way, most of the tasks are rapidly executed (short waiting times), and some ones experiment long waiting times. In the other hand, in a Poisson process where all tasks are treated equally FIFO (first in, first out), the expected times among events are more uniform [42]. The heavy tailed distribution is evident in the arrival of the vehicles to an intersection, caused by closed cross intersection priority introduced because of the traffic lights, or because of the human decision and the priority of the routes in the adjacent intersections. The stochastic process used to model the arrival of vehicles (events) to the intersection has a Pareto distribution, as shown in Figures 4.1 *c)* and *d)*. Figure 4.1 shows the difference between two stochastic processes, where the expected time between events $E\{\tau\}$ for both distributions are equal. It can also be appreciated that a Poisson distribution does not adjust to the behavior of the traffic flow, in contrast with the Pareto distribution, which approximates appropriately.

4.1 Experiment I

The intersection is controlled with the *Honey Bee Social Foraging* algorithm when the arrivals to the rows q_i are constants. There are two possible scenarios when the arrivals are constants, and depend on the saturation flow.

If the input flow is less than the saturation flow, there is a solution that minimizes $\sum E\{q_i(k)\}$, $k = 1, \dots, N_e$. The queues will be completely free, so these ones can be full again. i.e., they have a bounded suitability. So, there is a solution for the time phases that make $q_i \leq \mathcal{B}$ for all i , i.e., bounded by an upper value \mathcal{B} . The existence of the solution is intuitive: if the amount of vehicles that arrive to the intersection is less than the total capacity, the row will get empty. Figure 4.2 shows the behavior of the controlled system. It can be seen that the rows get empty, which allows to have

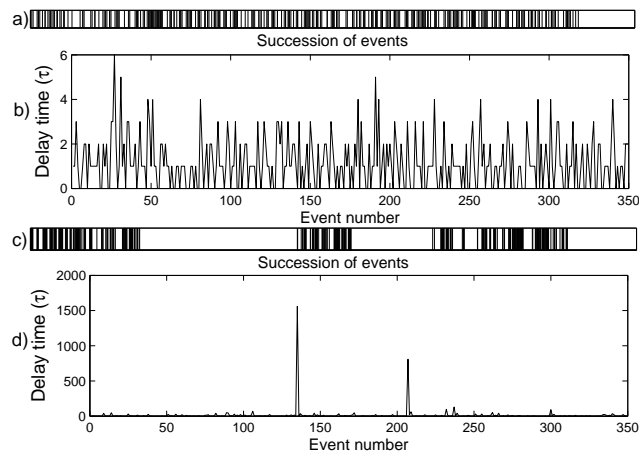


Figure 4.1: Differences in a stochastic process between a Poisson distribution and a heavy tailed distribution. **a)** Event distributions over time according to a process that obey to a Poisson distribution, where the event can take place in any moment with equal probability. The horizontal axis represents time, and each vertical line corresponds to an event. Notice that the expected times among events are similar and there are not big waiting times. **b)** Expected times for 300 consecutive events. The expected time of each event means one blank space in *a*). **c)** Event distribution over time for a heavy tailed distribution. **d)** The expected times for 300 consecutive events in a heavy tailed distribution. Notice the big peaks on the picture. These peaks correspond to big wait times, as shown in *c*). **b)** and **d)** are drawn in different scale because of the difference in the maximum values of the expected time. However, the two distribution have the same expected waiting time $\lambda = 6$ [4], [5].

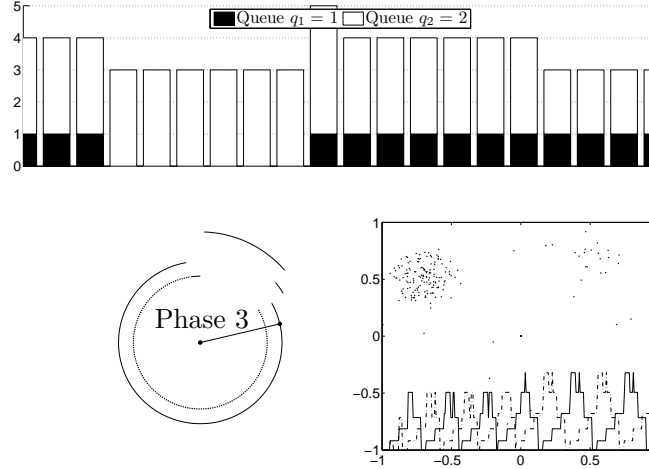


Figure 4.2: Intersection controlled using the *Honey Bee Social Foraging* algorithm when the vehicle arrivals are constant. Flow below the saturation point. On top the queue levels for a short simulation time is shown. The profitability function and the position for each bee are plotted in the bottom right sub plot. For the profitability function, the x axis is related to the time (from -1 to $+1$, where $+1$ is the full simulation time and -1 is the start simulation time). On the lower left corner, the clock of the phases is shown. In spite of the fact that the intersection has two stages, in the simulation, we have waiting time phases (phase 2 and phase 4), the phases that get the right of way are 1 and 3.

full rows after a while. On the profitability function versus time plot (Figure 4.2 at bottom right corner), the profitability J_f is shown for each zone (each one as a function of q_1 and q_2). In order to fit the profitability function in the subplot, an offset (-1) has been added to the profitability. Then, the profitability function value has zero as the maximum value, and -1 as the minimum value. In the same plot, the bee allocation in a 2×2 space is shown. If we observe the profitability functions for each zone in the last moment ($+1$), the profitability function from one zone is bigger than the other. This is because there are more vehicles in q_2 . Therefore, more bees are assigned to zone 2 (located in $[-0.7 \ 0.5]$) than to zone 1 (located in $[0.5 \ 0.7]$).

In Figure 4.2 it can be seen that the profitability functions are cyclical and their amplitude after some time gets bounded. Also, the function frequencies $q_2 = 2$ are stable after a transition time. In this way the phases after a time have the same length, so the relation vehicle/time is equal for the two phases. As in the IFD, where the suitability for each animal gets equal, the green time efficiency gets equal and becomes stable for the two phases (i.e., phases 1 and 3).

On the other hand, if the input flow is bigger than the saturation flow, there is no solution for the traffic problem [1], i.e., q_1 and q_2 are constantly growing, as well as the profitability function (until the saturation). In this case, the queues can never be completely empty and will grow closer to their profitability function (until the saturation). As we can see in the Figure 4.3. However, there is no reason why the IFD does not emerge. When the intersection is saturated there is not control strategy that gets the queues completely free. In Figure 4.3 c) the rows grow constantly because more vehicles arrive than those that the intersection can serve. When this occurs, the profitability functions for the two zones get saturated. For this reason, the same number of bees in the two zones and the phases time gets equal, as its shown in Figures 4.3 a) and d). At this moment, the intersection serves vehicles at maximal capacity, and since the vehicles are “unlimited” (at least for the maximum green time permitted), the performance at both phases (vehicles per time unit) remains the same.

In Figure 4.3 b) the suitability function for each site is shown. The suitability function $\frac{J_{f_i}}{x_i}$ for each site converges to the same value. This reflects the ideal distribution of the resources (green time) over all sites or phases, i.e., the relation between vehicles served and green time allocated (performance) gets equal for all routes. For this reason, the IFD emerges also under saturation conditions.

4.2 Experiment II

In this experiment, the intersection is controlled by a fixed time strategy, where random arrivals follow a Pareto distribution (heavy tailed). The design parameter is the expected time between events $E\{\tau\}$. Using this parameter, the arrivals are considered as constants. The optimum time for the two phases are calculated (because the stochastic model is equal for the two routes, the time for the two phases is equal too). For this event rate, the intersection is in a no saturation state (on average). Due to the nature of the stochastic process in some times of the simulation, the number of events gets higher in a few sampling times (expected times between events are short). As we can see in Figure 4.4 c). This models a characteristic from this type of process, where the congestion is caused for an excessive temporal and local demand. In the Figure 4.4 d) it is shown the sum of

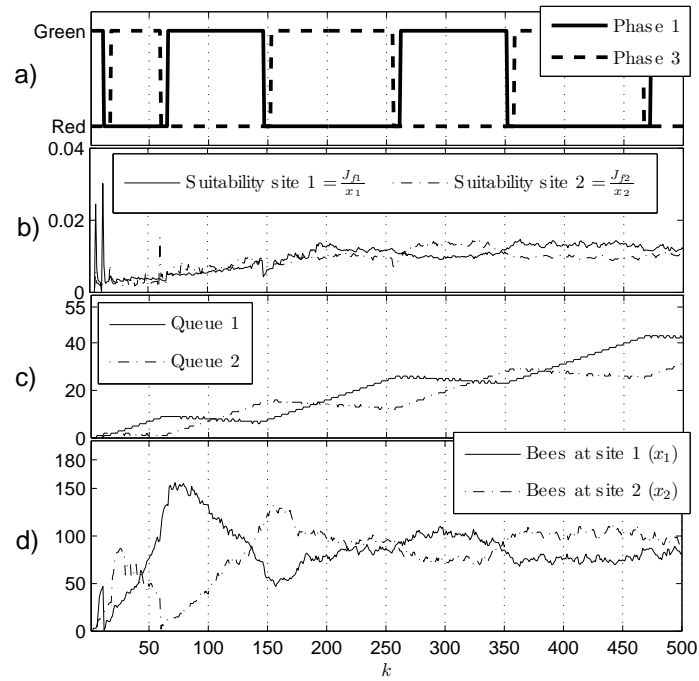


Figure 4.3: The x axes correspond to the simulation time (k). In **a)**, the state of each phase is shown. In **b)**, the suitability for each site is shown. Notice how the IFD is reached when the suitability functions converges. In **c)**, the value for the queues are plotted. Here, we can see why the intersection its over a saturation point. The queues are filled more faster than their full capacity can serve (i.e., r. o. w). In **d)**, The total number of bees at sites are shown. Notice how the IFD emerges when the suitability of sites 1 and 2 converges at a similar value.

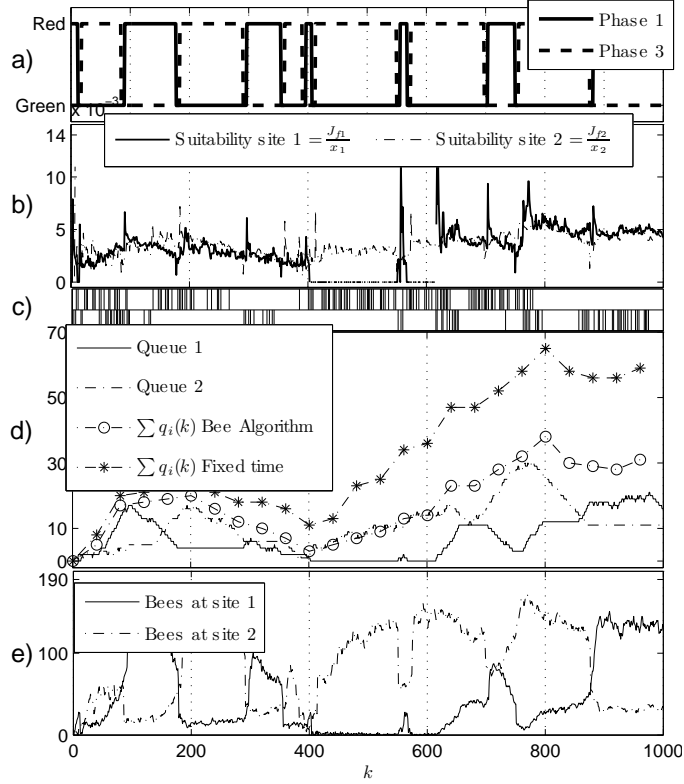


Figure 4.4: The x axes correspond to the simulation time (k). In any moment of the simulation, the fixed time strategy gives better results than the achieved for the bioinspired algorithm for the same sampling time.

the queues versus simulation time. Notice that the sum of the queues for the fixed time strategy is always bigger than in the algorithm proposed.

The state of the queues are shown on Figure 4.4 d). Just above at c) is the event register. The events are plotted as a function of the time (x axis). This plot is divided in two parts, each one related with one route of the intersection. Each vertical line represents the arrival of a vehicle for each queue.

4.3 Experiment III

In this experiment the intersection is controlled by the *Honey Bee Social Foraging* algorithm. As is shown in Figure 4.5. The performance of the controlled system by the bioinspired algorithm is higher. The size of the queues is significantly less as we can see in Figure 4.4 d), and with no sampling time in any Monte Carlo simulation, the relation between the performances of the two systems

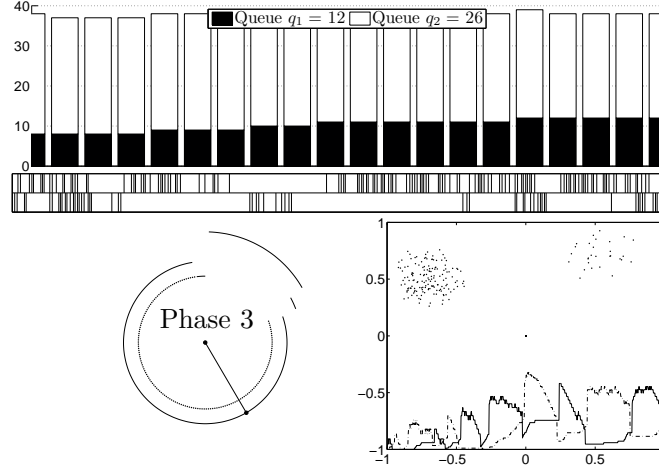


Figure 4.5: Intersection controlled using the *Honey Bee Social Foraging* algorithm when the vehicle arrivals follows a Pareto distribution (heavy tailed distribution). The time between events τ is exactly the same for the experiment *I*. The proposed solution presents a high adaptability to the circumstances, and the load among queues is stable.

gets better for the fixed control strategy.

In Figure 4.4 b) the suitability for the sites is shown. Also, in Figure 4.4 e) shows the bee allocation per site. As we can see, the suitability tries to converge at a common value for the two sites. The total number of bees at each site is different, but the suitability function shows that the allocation of the resources (green time) are balanced with respect to the load of the vehicles at the intersection. The IFD emerges at some intervals of the simulation. When the profitability of the patches (J_f) suddenly changes (vehicle arrivals or changes in the state of the semaphore), the foraging process tries to reach a new IFD. We can see the changes of the suitability according to the changes on the state of the semaphore in Figures 4.4 a) and b). Also, we can see the effect of the new vehicle arrivals c) in the suitability function. The foraging process tends to stabilize after some time, when the IFD is reached. For each Monte Carlo simulation, the average of the queues is calculated as a performance index of the two strategies. The histograms in Figure 4.6 illustrate that the average value of all expected values from the bioinspired algorithm is lower for the two lines. In the same way, the variance is lower in the algorithm proposed. Therefore, the implemented solution in this document has a better performance than the fixed control strategy (with an optimum configuration for this type of control).

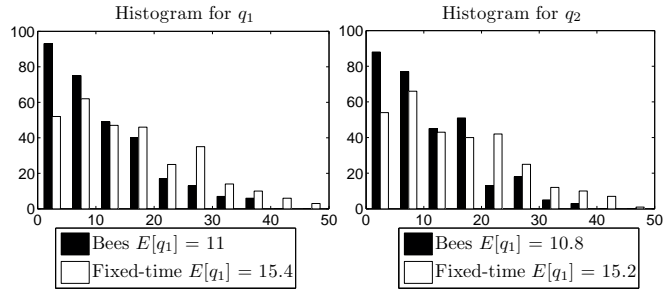


Figure 4.6: Statistics of the queues values with fixed time control and using social foraging. It shows clearly that the traffic responsive technique is superior.

4.4 Discussion

The bioinspired algorithm presents higher performance than a fixed time strategy. We have also proved that an ideal free distribution (IFD) emerges under a variety of scenarios, such as saturation and no saturation state, and also when the arrivals obey a stochastic process.

CHAPTER V

STABILITY ANALYSIS

In order to analyze the stability of the system (intersection and honey bee algorithm), we model the traffic dynamics as a ideal flow-fluid model [43], and the dynamics of the social bee foraging algorithm such as a replicator dynamics model [44][45][46].

5.1 Fluid-Flow model

Let q_i ([vehic/time]) be the vehicular flow, k_i ([vehic/long]) the vehicular density, and u_i ([long/time]) the mean speed at i^{th} link (as is shown in Figure 5.1). The vehicular flow is defined as $q_i \doteq k_i u_i$.

Using the fluid-flow analogy:

$$\begin{aligned} \frac{\partial k_i}{\partial t} + \frac{\partial q_i}{\partial l} &= 0 \\ \frac{\partial k_i}{\partial t} + \frac{u_i \partial k_i}{\partial l} + \frac{k_i \partial u_i}{\partial l} &= 0 \end{aligned} \quad (5.1)$$

It is a well established in theory of traffic flow that vehicular velocity varies inversely with the concentration of vehicles [43], i.e., $u_i = f(k_i)$, as consequence $\frac{\partial u_i}{\partial k} = \frac{\partial u_i}{\partial l} \frac{\partial l}{\partial k} = \frac{du_i}{dk} = u'_i$, $\frac{\partial u_i}{\partial l} = u'_i \frac{\partial k_i}{\partial l}$, substituting in Equation (5.1), results,

$$\begin{aligned} \frac{\partial k_i}{\partial t} + u_i \frac{\partial k_i}{\partial l} + k_i u'_i \frac{\partial k_i}{\partial l} &= 0 \\ \frac{\partial k_i}{\partial t} + (u_i + k_i u'_i) \frac{\partial k_i}{\partial l} &= 0 \end{aligned} \quad (5.2)$$

A well known generalized equation of motion is [43] $\frac{du_i}{dt} = -c_i^2 k_i^n \frac{\partial k}{\partial l}$, taking the total derivative:

$$\begin{aligned} \frac{du_i}{dt} &= \frac{\partial u_i}{\partial l} \frac{dl}{dt} + \frac{\partial u_i}{\partial t} \frac{dt}{dt} \\ \frac{du_i}{dt} &= u_i \frac{\partial u_i}{\partial l} + \frac{\partial u_i}{\partial t} \\ 0 &= u_i \frac{\partial u_i}{\partial l} + c_i^2 k_i^n \frac{\partial k}{\partial l} + \frac{\partial u_i}{\partial t} \end{aligned} \quad (5.3)$$

Also substituting $\frac{\partial u_i}{\partial t} = u'_i \frac{\partial k_i}{\partial t}$, and $\frac{\partial u_i}{\partial l} = u'_i \frac{\partial k_i}{\partial l}$ in Equation (5.3) results,

$$\begin{aligned} u'_i \frac{\partial k_i}{\partial l} u_i + c_i^2 k_i^n \frac{\partial k}{\partial l} + u'_i \frac{\partial k_i}{\partial t} &= 0 \\ \frac{\partial k_i}{\partial t} + \frac{\partial k_i}{\partial l} \left(u_i + \frac{c_i^2 k_i^n}{u'_i} \right) &= 0 \end{aligned} \quad (5.4)$$

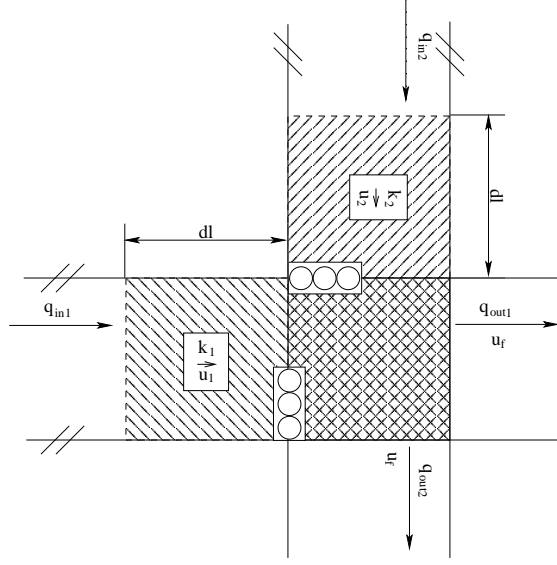


Figure 5.1: Vehicular input (q_{in_i}) and output (q_{out_i}) flows, densities (k_i), mean (u_i) and free speed (u_f) for a single intersection model.

which is the generalized equation of motion [43]. From Equations (5.2) and (5.4) we have $u'_i = -c_i k_i^{(n-1)/2}$. For $n \geq -1$ the solution is $u_i = -2c_i k_i^{(n+1)/2} / (n+1) + C_1$ [43]. There is a vehicular density where there are no movement (jam concentration k_{ij}) or $u_i = 0$. Solving for C_1 we get $C_1 = \frac{2c_i}{n+1} k_{ij}^{(n+1)/2}$ for $n \geq -1$. The speed for the i^{th} link is,

$$u_i = \frac{-2c_i}{n+1} \left(k_{ij}^{(n+1)/2} - k_i^{(n+1)/2} \right)$$

also where the density is equal zero ($k_i = 0$) then the drivers can travel at free speed $u_{if} = 2c_i k_{ij}^{(n+1)/2} / (n+1)$. Solving for $c_i = (n+1)u_{if} / (2k_{ij}^{(n+1)/2})$ which is a constant related to the capacity of the link. Replacing it in Equation (5.3) we get.

$$u_i = u_{if} \left(1 - \left(\frac{k_i}{k_{ij}} \right)^{(n+1)/2} \right) \quad (5.5)$$

$$q_i = k_i u_i = k_i u_{if} \left(1 - \left(\frac{k_i}{k_{ij}} \right)^{(n+1)/2} \right) \quad (5.6)$$

Re-writing the Equation (5.1),

$$k_i dl - \left(k_i - \frac{\partial k_i}{\partial t} dt \right) dl = q_i dt - \left(q_i + \frac{\partial q_i}{\partial l} dl \right) dt \quad (5.7)$$

where $k_i dl$ is the number of vehicles in the link i at $(l, l+dl)$ during the interval dt , $\left(k_i - \frac{\partial k_i}{\partial t} dt \right) dl$ is the change in the number of vehicles in dl caused by a change in the density ∂k_i during the

interval dt , $q_i dt$ is the number of vehicles entering to the link i at $(l, l + dl)$ during the interval dt , $\left(q_i + \frac{\partial q_i}{\partial t} dl\right) dt$ is the vehicles leaving $(l, l + dl)$. Re-writing the Equation (5.7)

$$\frac{dk_i}{dt} \alpha = q_{in_i} \beta - q_{out_i} \beta \quad (5.8)$$

where α and β are constants with the appropriate dimensions. For the single intersection problem q_{in_i} is a constant (demand), and q_{out_i} is a function of the green time assigned to the i^{th} link. In order to analyze the the dynamics of the system, it is assumed that the velocity in the link is (u_i) when the link has r.o.w. When the semaphore state is red, the velocity for that link is zero ($u_i = 0$). For this reason the velocity is the proportion of the green allocation time in each cycle. This is an approach similar to the one used in the *Store-and-Forward* strategies. This approach was first suggested by Gazis and Potts [47], [1] and has since been used in various works notably for road traffic control [48][49]. The main idea when using store-and-forward models for road traffic control is to introduce a model simplification that enables the mathematical description of the traffic flow process without use of discrete variables. This is of paramount importance because it opens the way to the application of a number of highly efficient optimization and control methods [3]. The critical simplification occurs when we model the outflow q_{out_i} of a stream according to its velocity

$$u_i = u_{if} \left(1 - \left(\frac{k_i}{k_{ij}} \right)^{(n+1)/2} \right) \frac{g_i}{c}$$

where g_i is the green time duration for this link and c is the cycle time. In Figure 5.2 is shown that u_i is equal to the average velocity during the corresponding cycle, rather than equal to u_i during the green phase and equal to zero during the red phase (like a pulse width modulation scheme). In other words, we model the flow as a continuous (uninterrupted) for each network link. The consequences of this type of simplifications are as follows [3]: (i), the sampling time of the discrete-time representation cannot be shorter than the cycle time c ; hence, real-time decisions cannot be taken more frequently than at every cycle; (ii), the oscillations of vehicle queues in the links due to green/red commutations are not described by de model; (iii), the effect of offset for consecutive intersections cannot be described by the model. Despite these consequences, the appropriate use of this models may lead to efficient coordinated control strategies for large-scale networks as demonstrated in simulation studies in some of the references in [3]. The allocation time per phase (g_i) depends on the

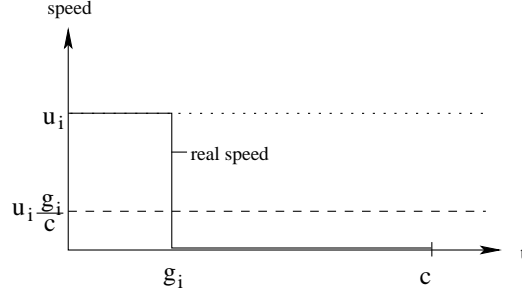


Figure 5.2: Simplified modeling of link speed u_i . Adapted from [3].

bee allocation per site ($B_i = x_i/B$), i.e.,

$$\frac{dk_i}{dt}\alpha = q_{in_i}\beta - q_{out_i}\beta\frac{x_i}{B}$$

replacing q_{out_i} from the Equation (5.6), and for a single intersection ($i = 1, 2$), such as the one shown in the Figure 5.1 and 2.3, the densities are:

$$\begin{aligned} \dot{k}_1\alpha &= q_{in_1}\beta - k_1u_{1f}\left(1 - \left(\frac{k_1}{k_{1j}}\right)^{(n+1)/2}\right)\beta\frac{x_1}{B} \\ \dot{k}_2\alpha &= q_{in_2}\beta - k_2u_{2f}\left(1 - \left(\frac{k_2}{k_{2j}}\right)^{(n+1)/2}\right)\beta\frac{x_2}{B} \end{aligned} \quad (5.9)$$

5.2 The Replicator Dynamics approach

The replicator dynamics are a simple model how selection via differential fitness affects the proportions of animals using different strategies [45], [46]. In [44], the animal fitness $f(i)$ is described as the number of nutrients per time unit that an animal gets at habitat i , also the analysis of the game-theoretic and optimality properties of the IFD using the replicator dynamics approach is shown; in the same way, the replicator dynamics used in this chapter are based in an non linear game a difference than the original definition.

The fitness function chosen for the replicator dynamics model is a function of the vehicular flow and the total number of animals allocated in the i^{th} link:

$$f_i = \frac{q_{in_i}}{S_i x_i}$$

The dynamics for the allocation per site are [44],

$$\dot{x}_i = x_i (f_i - \bar{f})$$

$$\dot{x}_i = x_i \left(\frac{q_{in_i}}{S_i x_i} - \sum_{j=1}^N \frac{x_j}{B} \frac{q_{in_j}}{S_j x_j} \right)$$

where N is the number of zones or sites. For a single intersection ($i = 1, 2$),

$$\dot{x}_i = \frac{q_{in_i}}{S_i} - \frac{x_i(q_{in_1}S_2 + q_{in_2}S_1)}{BS_1S_2}$$

5.3 System equilibrium points

For a single intersection (Figures 2.3 and 5.1) the equilibrium point for the bee allocation x_i and the link densities k_i can be calculated as follows.

$$x_i^* = \frac{q_{in_i}BS_1S_2/S_i}{q_{in_1}S_2 + q_{in_2}S_1} \quad (5.10)$$

For a single intersection ($i = 1, 2$),

$$x_1^* = \frac{q_{in_1}BS_2}{q_{in_1}S_2 + q_{in_2}S_1}, x_2^* = \frac{q_{in_2}BS_1}{q_{in_1}S_2 + q_{in_2}S_1}$$

then, the equilibrium point can be rewritten as:

$$\frac{q_{in_1}}{S_1 x_1^*} = \frac{q_{in_2}}{S_2 x_2^*} \quad (5.11)$$

The Equation (5.11) is the optimum point proposed for Denos C. Gazis in 1965 [50] as a well known empirical formula. Replacing Equation (5.10) in Equation (5.9) and solving for the equilibrium point $k_i \alpha = 0$

$$q_{in_i} - k_i^* u_{if} \left(1 - \left(\frac{k_i^*}{k_{ij}} \right)^{(n+1)/2} \right) \frac{x_i^*}{B} = 0 \quad \text{for } n = 1$$

if $q_{in_i} \geq 0$

$$k_i^{*2} \frac{u_{if}S_1S_2/S_i}{k_{ij}(q_{in_1}S_2 + q_{in_2}S_1)} - k_i^* \frac{u_{if}S_1S_2/S_i}{(q_{in_1}S_2 + q_{in_2}S_1)} + 1 = 0$$

$$k_{i1,2}^* = \frac{k_{ij}}{2} \pm \frac{S_i k_{ij}}{2\zeta} \sqrt{\zeta^2 - \frac{4\zeta}{S_i k_{ij}}}$$

Where $\zeta = \frac{u_{if}S_1S_2}{q_{in_1}S_2 + q_{in_2}S_1}$

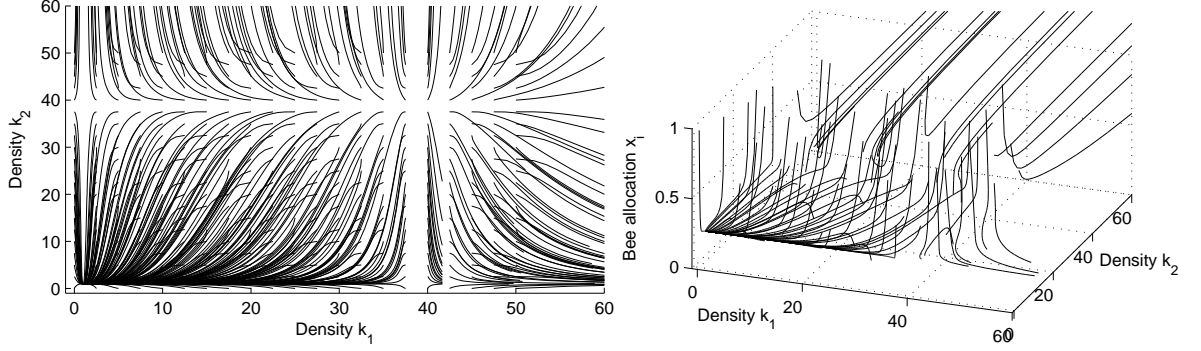


Figure 5.3: Trajectories of the system for some initial conditions.

5.4 Phase plane

According to the previous part, we can analyze the system trajectories (bee allocation x_i and densities k_i) for a single intersection problem using a phase plane. For some particular single intersection (as the one studied in Section 3.2, Figures 2.3 and 5.1. Additional results can be found in [51] and [52]), let $q_{in_1} = 1/21$ and $q_{in_2} = 1/7$ [vehic/instant] be the input vehicular flow, $s_1 = s_2 = 1/5$ be the saturation flow, $k_{1j} = k_{2j} = 40$ [vehic/distance] be the jam density, $u_{1f} = u_{2f} = 1/5$ [distance/instant] be the free speed. In the Figure 5.3 the trajectories for some initial conditions are shown.

In Figure 5.3, we can see how the trajectories convey to an equilibrium point at $k_{i_1}^* = 0.9762$ when the initial conditions are $k_{1,2}(0) < k_{i_2}^*$ where $k_{i_2}^* = 39.0238$.

5.5 Discussion

At the equilibrium point (located in $k_{i_1}^* = 0.9762$), the resource allocation (bee allocation) is according to the IFD for the replicator dynamics model and the evolutionary algorithm (as we can see in Figures 5.3 and 5.4). The *Honey Bee Social Foraging* algorithm (studied in Section 3.2) allocate its resources in the same way that the replicator dynamics model (the percentage of time used for some phase in the entire cycle respect to the bee allocation x_i); for this reason, the evolutionary algorithm convey to the optimum point proposed in [50] (Equation (5.11)) and this optimum point is function of the input flows of the intersection.

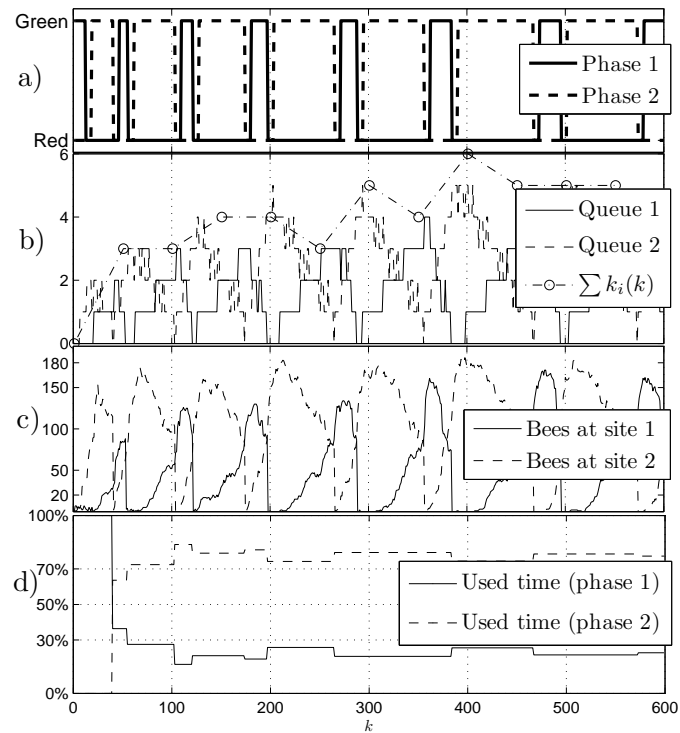


Figure 5.4: Simulation for a single intersection controlled using the *Honey Bee Social Foraging* algorithm. **a)** The states of each phases. **b)** The queue state for each approach. **c)** The bee allocation (resource allocation). **d)** The time used for each phase per cycle.

CHAPTER VI

MULTI-HIVE AND MULTI-SITE PROBLEM

A large scale control problem in transportation such as controlling the traffic signals in a traffic network can be simplified into small networks in order to analyze the properties of the strategy used. In this section, we propose a small network signalization problem. The network used in this approach are symmetrical, and the intersections are similar to the one shown in Figure 2.3; but, the speeds and densities in the links are coupled with the speeds and densities of adjacent links. The control strategy used to control the network (simulated in VISSIM) is similar to the one studied in previous sections. In this case, there are the possibility of have a multiple hive game, where all the bees of each hive try to increase its own profitability harvesting in a multiple site environment. The multi-hive game are simulated in Matlab (as in previous sections), the hives have the same functional structure as the one shown in Section 3.2. The main difference is that the habitat is degrading in suitability via the visit of each additional bee of multiple hives in the environment rather than a single one. In some simulation scenarios, the hives do not have the freedom (or knowledge) for harvesting in all sites in the habitat. This scenarios are included in order to depict the distributed computational cost in large scale traffic control systems. As in previous sections (Sections 4, 4.1, 4.2 and 4.3), the traffic variables such as queues, density or vehicular speed for the links represent the suitability for each site. For this reason there are two sites per intersection as is shown in Figure 6.2. The traffic network is simulated using VISSIM with the same structure as the one shown in Figure 6.2. VISSIM is a microscopic, time step and behavior based simulation model developed to model urban traffic and public transit operations. VISSIM can analyze traffic and transit operations under constraints such as lane configuration, traffic composition, traffic signals, transit stops, etc. [53]. Using the COM (component object model) interface of VISSIM it is easy to link the microscopic traffic simulation with the evolutionary algorithm running in Matlab as we can see in Figure 6.1.

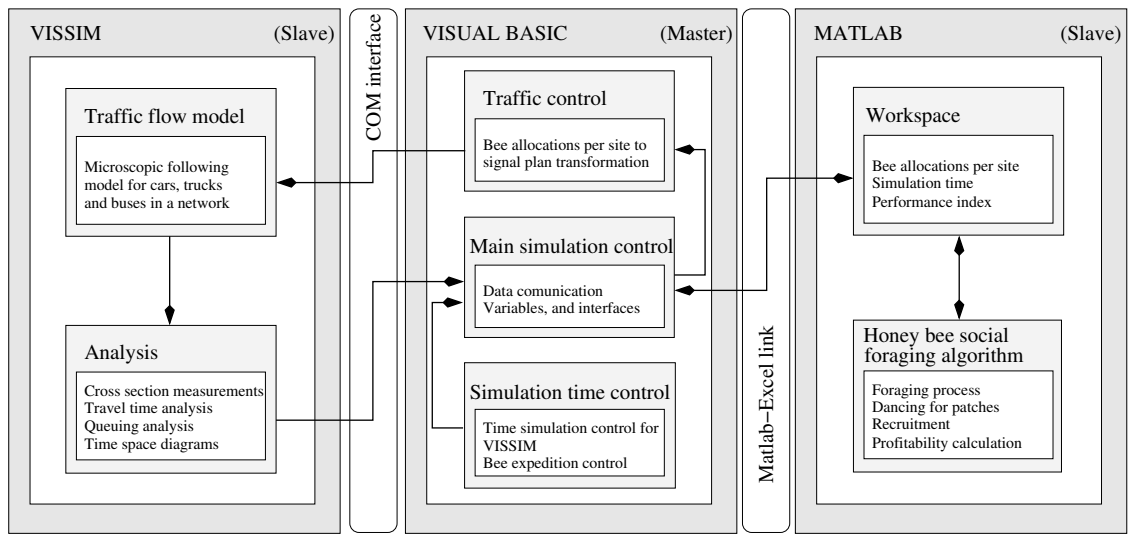


Figure 6.1: VISSIM - Visual Basic - Matlab bridge.

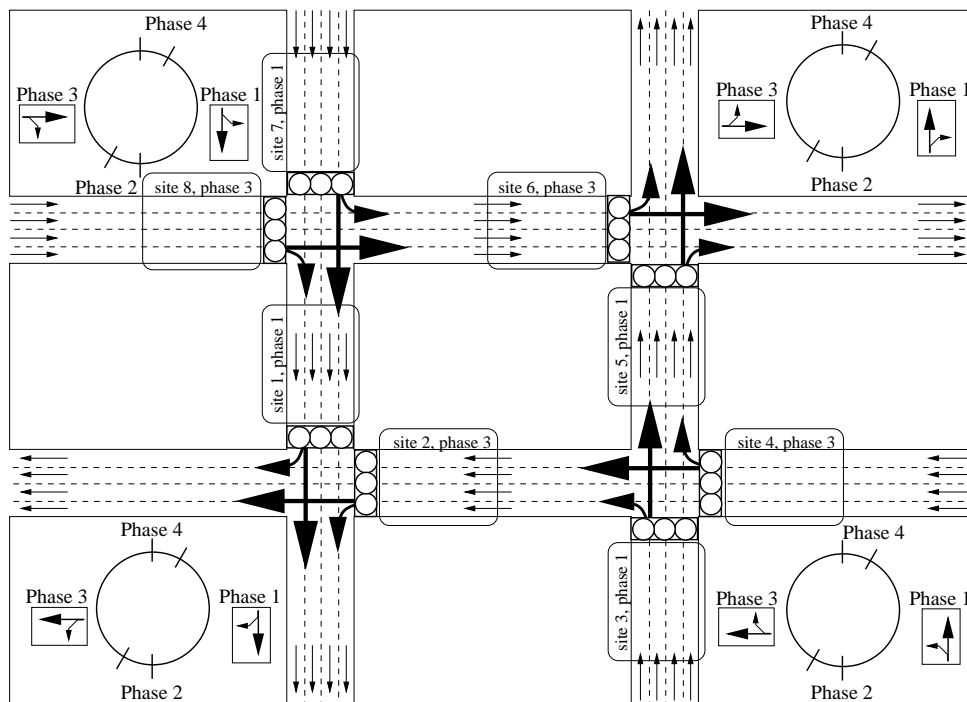


Figure 6.2: Multi-site and multi-hive simulation scheme.

6.1 Implementation

The evolutionary algorithm is used to control a small network simulated in VISSIM. The network has four nodes (intersections), each of these are similar to the one shown in Figure 2.3. The signal controllers in VISSIM (one per intersection) are commanded by the bee allocation in the multi hive game (simulated in Matlab); in the same way, the traffic variables (densities and vehicular speed) are simulated in VISSIM and transmitted to the *Honey Bee Social Foraging* Algorithm through a Visual Basic link (using COM Interface of VISSIM and Matlab-Excel Link of Matlab) as a feedback for the traffic control system. In Figure 6.1 is shown the structure and information flow of the overall simulation. The simulation has a master-slave scheme. The algorithm in Visual Basic plays the master role; here, the simulation time control for the *Honey Bee Social Foraging* Algorithm and the microscopic traffic simulation determines when a new expedition and simulation instant (respectively) occurs. The main simulation control for the simulation receives and transmits the traffic variables and the bee allocation according to the simulation time control module. The bee allocation is transformed into a signal plan according to VISSIM signal controllers, this plan is set out via COM interface each simulation time; then, a new simulation time is evaluated by VISSIM. The microscopic traffic simulator, based on the car following model developed by Wiedemann [30] determines the position and velocity of each vehicle in the network; later, in the analysis module the traffic variables are calculated and transmitted to the Visual Basic algorithm via COM interface. The Main Simulation Control (in Visual Basic) writes the traffic variables into the Matlab workspace and executes a new bee expedition (*Honey Bee Social Foraging* algorithm simulation instant); then, the new bee allocations are returned using again the Matlab-Excel Link.

In Figure 6.2 the network simulated in VISSIM is shown. Each intersection has an independent controller, four phases and two possible r.o.w configurations (phase 1 and phase 3). The sites for the multi-hive game are distributed as shown in Figure 6.2.

Several scenarios are evaluated in this simulation. The parameters changed in these scenarios are: Traffic composition, decision routing and traffic variable selection for suitability function. The

traffic composition are related with the vehicle input that obey a stochastic process for the links entering to the network. These links are namely as follows: the upper left horizontal link, upper left vertical link, lower right horizontal link and lower right vertical link as (1), (2), (3) and (4) respectively. The decision routing refers to the probability of that a given vehicle turn in some specific intersection. According to this simulation parameters, eight scenarios are evaluated as follows: (i), (1) and (3) equal to 3200 *veh/h*, (2) and (4) equal to 1600 *veh/h*, the decision route as (50 – 50)%, and the traffic variable as the vehicular density k_i ; (ii), (1) and (3) equal to 3200 *veh/h*, (2) and (4) equal to 1600 *veh/h*, the decision route as (50 – 50)%, and the traffic variable as the vehicular speed u_i ; (iii), (1), (2), (3) and (4) equal to 3200 *veh/h*, the decision route as (50 – 50)%, and the traffic variable as the vehicular density k_i ; (iv), (1), (2), (3) and (4) equal to 3200 *veh/h*, the decision route as (50 – 50)%, and the traffic variable as the vehicular speed u_i ; (v), (1) and (3) equal to 3200 *veh/h*, (2) and (4) equal to 1600 *veh/h*, the decision route as (80 – 20)% (horizontal - vertical preference), and the traffic variable as the vehicular density k_i ; (vi), (1) and (3) equal to 3200 *veh/h*, (2) and (4) equal to 1600 *veh/h*, the decision route as (80 – 20)% (horizontal - vertical preference), and the traffic variable as the vehicular speed u_i ; (vii), (1), (2), (3) and (4) equal to 3200 *veh/h*, the decision route as (80 – 20)% (horizontal - vertical preference), and the traffic variable as the vehicular density k_i ; (viii), (1), (2), (3) and (4) equal to 3200 *veh/h*, the decision route as (80 – 20)% (horizontal - vertical preference), and the traffic variable as the vehicular speed u_i . The scenarios are summarized in Table 2. The performance indexes evaluated for the eight scenarios are: Average delay time per vehicle, average number of stops per vehicles, average speed, total delay time and total travel time.

6.2 Results and discussion

In Table 3 the simulation results are summarized. For the same conditions (decision routing and traffic composition) the best results are achieved when the traffic variable is k_i than u_i . The performance of minimize the density with respect to maximize the speed is superior in all the performance indexes. The traffic composition for the (3200 – 1600)[*veh/h*] scheme is the 75% of the total amount of vehicles entering to the network for the (All 3200)[*veh/h*] scheme; nonetheless, the

Table 2: Simulation scenarios.

Simulation scenario	Decision Routing [%]	Traffic composition (scheme) [veh/h]	Traffic variable
(i)	(50 – 50)	(3200 – 1600)	k_i
(ii)	(50 – 50)	(3200 – 1600)	u_i
(iii)	(50 – 50)	(All 3200)	k_i
(iv)	(50 – 50)	(All 3200)	u_i
(v)	(80 – 20)	(3200 – 1600)	k_i
(vi)	(80 – 20)	(3200 – 1600)	u_i
(vii)	(80 – 20)	(All 3200)	k_i
(viii)	(80 – 20)	(All 3200)	u_i

Table 3: Simulation results.

Simulation scenario	Average delay time per vehicle [s]	Average number of stops per vehicles	Average speed [km/h]	Total delay time [h]	Total travel time [h]
(i)	24.612	0.733	37.722	64.888	91.256
(ii)	43.144	2.386	24.339	112.906	138.787
(iii)	44.085	1.604	23.652	154.859	189.103
(iv)	68.660	5.092	16.017	203.941	232.421
(v)	22.259	0.591	43.047	58.682	87.550
(vi)	25.700	0.713	39.009	67.756	96.644
(vii)	36.382	1.230	29.661	127.922	165.503
(viii)	52.229	3.309	22.015	168.830	203.035

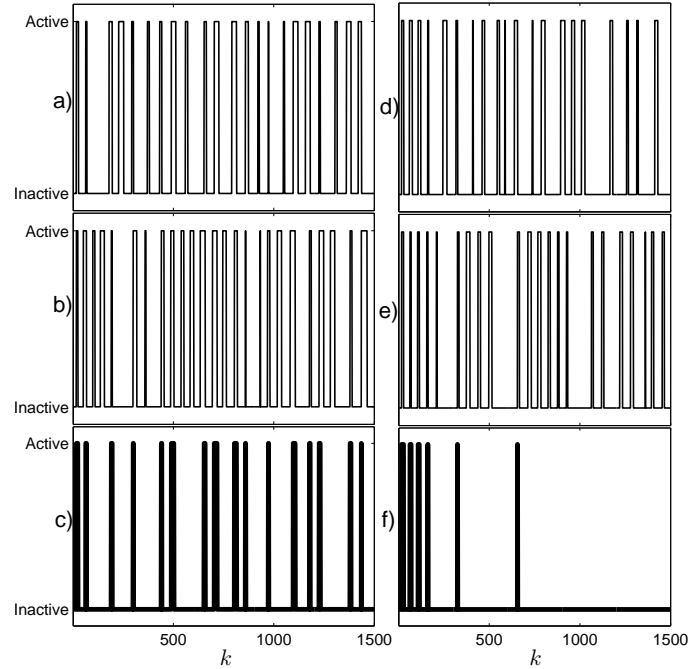


Figure 6.3: Green waves that emerges naturally in the evolutionary algorithm. a), b) and c) are simulation results for the *(vii)* scenario, and d), e) and f) are simulation results for the *(iii)* scenario. **a)** and **d)** shown the green waves in the lower horizontal link (when the 3^{th} phase for the lower right and left controllers are active at the same time). **b)** and **e)** shown the green waves in the upper horizontal link (when the 3^{th} phase for the upper right and left controllers are active at the same time). **c)** and **f)** shown the instants when the green waves in both directions (right to left and left to right) are active (when the 3^{th} phase for all controllers are active at the same time).

performance indexes drops dramatically (up to 50%) when the flow is increase only 25%, this is because the network is near to the theoretical saturation point. We can see that the better performance of the selection of k_i remains. When we compare the two decision routes alternatives, we can see that the performance is better (in all performance indexes) in the (80 – 20)% scenarios for the same conditions. This is because the hives, in order to increase its profitability, allocate more resources in the places more suitable. When the horizontal links are more attractive (suitable) the overall performance is increased. In addition, more green waves emerge naturally for the same conditions when the horizontal links have preference, as is shown in Figure 6.3. In this figure the simulation results for the *(iii)* and *(vii)* scenarios are shown. Also, is shown how if there is a preference in the horizontal links more green waves emerges.

In order to estimate the relation between the traffic signals; in Figure 6.4, is shown the cross

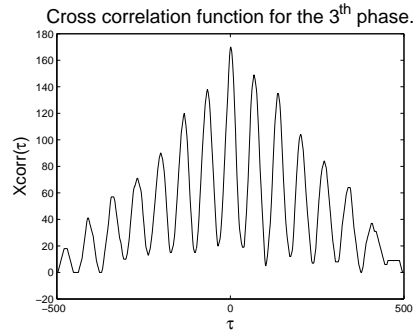


Figure 6.4: Cross correlation function for the 3th phase in the upper left and lower right intersection for the simulation results of the (vii) scenario.

correlation function for the 3th phase in the upper left and lower right intersection. This two intersections not are directly linked, but, the traffic signals are heavily related in a periodically manner, as its shown in the cross correlation function.

CHAPTER VII

CONCLUSIONS AND FUTURE WORK

In this work we describe the analogies between the green time allocation problem and the *Honey Bee Social Foraging* problem. Based on these analogies and in the work by [14]-[15] a novel traffic control strategy for a single intersection is proposed. The bioinspired algorithm presents higher performance than a fixed time strategy. We have also proved that an ideal free distribution (IFD) emerges under a variety of scenarios, such as saturation and no saturation state, and also when the arrivals obey a stochastic process.

A model using the replicator dynamics approach for the *Honey Bee Social Foraging* Algorithm and a fluid flow model for the traffic flow is proposed. The equilibrium points of the system for the resource allocation are the same in the evolutionary algorithm and in the model proposed. It is shown how the optimal point proposed by Gazis and Edie (1968)[50] is reached by the evolutionary algorithm and by the model.

We propose a small network signalization problem. The network (simulated in VISSIM) is controlled using the evolutionary algorithm in a Multi-Hive and Multi-Site environment. Several scenarios are simulated for the network proposed, in each of these scenarios, parameters such as decision routing and traffic composition are evaluated. It is shown how green waves emerges naturally in the system and how this affect with the overall performance.

The promising results obtained lead to a future implementation on large traffic networks, where a multi-hive game could be analyzed more extensively.

REFERENCES

- [1] D. Gazis, *Traffic Theory*. Kluwer Academic Publishers, 2002.
- [2] A. Davol, “Modeling of Traffic Signal Control and Transit Signal Priority Strategies in a Microscopic Simulation Laboratory,” Ph.D. dissertation, Massachusetts Institute of Technology, 2001.
- [3] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang, “Review of road traffic control strategies,” *Proceedings of the IEEE*, vol. 91, no. 12, pp. 2043–2067, 2003.
- [4] A. Barabasi, “The origin of bursts and heavy tails in human dynamics,” *Nature(London)*, vol. 435, no. 7039, pp. 207–211, 2005.
- [5] G. Oliveira and A. Barabasi, “Human dynamics: the correspondence patterns of Darwin and Einstein, 2005,” *Nature*, vol. 437, p. 1251.
- [6] N. Gartner, T. R. Board, and N. R. C. (US), *OPAC: A Demand-responsive Strategy for Traffic Signal Control*. Transportation Research Board, National Research Council, 1983.
- [7] J. Henry, J. Farges, and J. Tuffal, “The PRODYN real time traffic algorithm,” *Proc. of the IFAC Symposium, Baden-Baden*, 1983.
- [8] S. SEN and K. HEAD, “Controlled optimization of phases at an intersection,” *Transportation science*, vol. 31, no. 1, pp. 5–17, 1997.
- [9] K. Passino, *Biomimicry for Optimization, Control, and Automation*. Springer, 2005.
- [10] M. Dorigo and C. Blum, “Ant colony optimization theory: A survey,” *Theoretical Computer Science*, vol. 344, no. 2-3, pp. 243–278, 2005.
- [11] R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz, “Ant-Based Load Balancing in Telecommunications Networks,” *Adaptive Behavior*, vol. 5, no. 2, p. 169, 1997.

- [12] M. Reimann, K. Doerner, and R. Hartl, “D-Ants: Savings Based Ants divide and conquer the vehicle routing problem,” *Computers and Operations Research*, vol. 31, no. 4, pp. 563–591, 2004.
- [13] D. Teodorovic and M. Dell’Orco, “Mitigating Traffic Congestion: Solving the Ride-Matching Problem by Bee Colony Optimization,” *Transportation Planning and Technology*, vol. 31, no. 2, pp. 135–152, 2008.
- [14] N. Quijano and K. Passino, “Honey Bee Social Foraging Algorithms for Resource Allocation, Part I: Algorithm and Theory,” *In Proceedings of the 2007 American Control Conference. ACC’07*, pp. 3383–3388, 2007.
- [15] —, “Honey Bee Social Foraging Algorithms for Resource Allocation, Part II: Application,” *In Proceedings of the 2007 American Control Conference. ACC’07*, pp. 3389–3394, 2007.
- [16] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, USA, 1999.
- [17] H. Ceylan and M. Bell, “Traffic signal timing optimisation based on genetic algorithm approach, including drivers’ routing,” *Transportation Research Part B*, vol. 38, no. 4, pp. 329–342, 2004.
- [18] D. Srinivasan, W. Loo, and R. Cheu, “Traffic incident detection using particle swarm optimization,” *Swarm Intelligence Symposium, 2003. SIS’03. Proceedings of the 2003 IEEE*, pp. 144–151, 2003.
- [19] S. Fretwell and H. Lucas, “On territorial behavior and other factors influencing habitat distribution in birds. I. Theoretical development,” *Acta Biotheoretica*, vol. 19, no. 1, pp. 16–36, 1970.
- [20] H. de Vries and J. Biesmeijer, “Self-organization in collective honeybee foraging: emergence of symmetry breaking, cross inhibition and equal harvest-rate distribution,” *Behavioral Ecology and Sociobiology*, vol. 51, no. 6, pp. 557–569, 2002.

- [21] J. Bartholdi, T. Seeley, C. Tovey, and J. Vande Vate, “The pattern and effectiveness of forager allocation among flower patches by honey bee colonies,” *Journal of theoretical biology*, vol. 160, no. 1, pp. 23–40, 1993.
- [22] R. Dukas and L. Edelman-Keshet, “The Spatial Distribution of Colonial Food Provisioners,” *Journal of Theoretical Biology*, vol. 190, no. 2, pp. 121–134, 1998.
- [23] S. Nakrani and C. Tovey, “From honeybees to Internet servers: biomimicry for distributed management of Internet hosting centers,” *Bioinspiration and Biomimetics*, vol. 2, no. 4, pp. S182–S197, 2007.
- [24] R. Walker, T. Inc, and C. Los Angeles, “Emulating the honeybee information sharing model,” *Integration of Knowledge Intensive Multi-Agent Systems, 2003. International Conference on*, pp. 497–504, 2003.
- [25] T. Seeley, *The Wisdom of the Hive: The Social Physiology of Honey Bee Colonies*. Harvard University Press, 1995.
- [26] R. Hall, *Handbook of Transportation Science*. Kluwer Academic Publishers, 2003.
- [27] M. Lighthill and G. Whitham, “On Kinematic Waves. I. Flood Movement in Long Rivers,” *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences (1934-1990)*, vol. 229, no. 1178, pp. 281–316, 1955.
- [28] ———, “On Kinematic Waves. II. A Theory of Traffic Flow on Long Crowded Roads,” *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences (1934-1990)*, vol. 229, no. 1178, pp. 317–345, 1955.
- [29] S. Hoogendoorn and P. Bovy, “State-of-the-art of vehicular traffic flow modelling,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 215, no. 4, pp. 283–303, 2001.
- [30] R. Wiedemann, “Simulation des Straßenverkehrsflusses [Traffic Flow Simulations],” *Publication Series of the Institute for Traffic (Karlsruhe: University of Karlsruhe)*, 1974.

- [31] R. Allsop, "SIGSET: A computer program for calculating traffic signal settings," *Traffic Engineering and Control*, vol. 12, no. 2, 1971.
- [32] —, "SIGCAP: A computer program for assessing the traffic capacity of signal-controlled road junctions," *Traffic Engineering & Control*, vol. 17, pp. 338–341, 1976.
- [33] F. Webster, *Traffic Signal Settings*. HM Stationery Off., 1958.
- [34] W. Wey, "Model formulation and solution algorithm of traffic signal control in an urban network," *Computers, Environment and Urban Systems*, vol. 24, no. 4, pp. 355–378, 2000.
- [35] D. Robertson, "TRANSYT method for area traffic control," *Traffic Engineering and Control*, vol. 11, no. 6, pp. 276–281, 1969.
- [36] J. Little, "The Synchronization of Traffic Signals by Mixed-Integer Linear Programming," *Operations Research*, vol. 14, no. 4, pp. 568–594, 1966.
- [37] R. VINCENT and C. YOUNG, "Self-optimising traffic signal control using microprocessors. The TRRL MOVA strategy for isolated intersections," *Traffic engineering & control*, vol. 27, no. 7-8, pp. 385–387, 1986.
- [38] M. Dotoli, M. Fanti, and C. Meloni, "A signal timing plan formulation for urban traffic control," *Control Engineering Practice*, vol. 14, no. 11, pp. 1297–1311, 2006.
- [39] P. Hunt, R. Bretherton, D. Robertson, and M. Royal, "SCOOT on-line traffic signal optimisation technique," *Traffic Engineering and Control*, vol. 23, pp. 190–2, 1982.
- [40] A. Al-Mudhaffar, *Impacts of traffic signal control strategies*. Division of transports and logistics, Royal Institute of Technology.
- [41] F. Boillot, J. Blosseville, J. Lesort, V. Motyka, M. Papageorgiou, and S. Sellam, "Optimal signal control of urban traffic networks." *Road Traffic Monitoring, 1992 (IEE Conf. Pub. 355)*, 1992.
- [42] A. Barabasi, "The Architecture of Complexity," *Control Systems Magazine, IEEE*, vol. 27, no. 4, pp. 33–42, 2007.

- [43] D. Drew, *Traffic Flow Theory and Control*. McGraw-Hill, 1968.
- [44] N. Quijano and K. Passino, “The Ideal Free Distribution: Theory and Engineering Application,” *Systems, Man and Cybernetics, Part B, IEEE Transactions on*, vol. 37, no. 1, pp. 154–165, 2007.
- [45] J. Hofbauer and K. Sigmund, *Evolutionary Games and Population Dynamics*. Cambridge University Press, 1998.
- [46] J. Weibull, *Evolutionary Game Theory*. MIT Press, 1995.
- [47] D. Gazis and R. Potts, “The oversaturated intersection,” in *Proceedings 2nd Int. Symp. Traffic Theory*. Organisation for Economic Co-operation and Development, 1965, p. 221–237.
- [48] D. Gazis, “Optimum Control of a System of Oversaturated Intersections,” *Operations Research*, vol. 12, no. 6, pp. 815–831, 1964.
- [49] K. Kim and M. Bell, “Development of an integrated traffic control strategy for both urban signalised and motorway networks,” in *Proc. 1st Meeting EURO Working Group on Urban Traffic and Transportation, Landshut, Germany*, 1992.
- [50] D. Gazis and L. Edie, “Traffic flow theory,” *Proceedings of the IEEE*, vol. 56, no. 4, pp. 458–471, 1968.
- [51] P. Ñañez and N. Quijano, “Forraje Social en Control de Tráfico Urbano,” *IEEE Colombian Workshop on Robotics and Automation CWRA-2008*, 2008.
- [52] —, “Honey Bee Social Foraging for Urban Traffic Control,” *Submitted for conference publication, American Control Conference (ACC) 2009*, 2008.
- [53] P. AG, *VISSIM 4.10 User Manual*, 2005.